Verifying Adversarial Robustness of 3D Object Detectors for Autonomous Vehicles

Rebecca Dollahite[1*], Kevin Wang[1*], Kaidong Li[2], Yiqing Zhang[1], Ziming Zhang[3]
[1]Department of Data Science, Worcester Polytechnic Institue, MA, USA
[2]Department of EECS, University of Kansas, KS, USA
[3]Department of ECE, Worcester Polytechnic Institue, MA, USA
{rdollahite1, kwang14, yzhang37, zzhang15}@wpi.edu, kaidong.li@ku.edu

2022 IEEE MIT Undergraduate Research Technology Conference (URTC) | 978-1-6654-7345-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/URTC56832.2022.10002253

***Abstract*—Leading 3D object detectors for automated vehicles, such as PIXOR, do not robustly account for noise and are vulnerable to adversarial attacks. Existing attack methods do not accurately simulate naturally occurring noise, as they attempt to continuously on a discrete input space. In this paper, we propose a novel attack method, which maximizes loss by making gradient-informed, discrete changes. A subset of points within an image move based on a percentage change between the original and new gradient. We measure the validity of an attack based on its visual similarity to the original point cloud and numeric metrics.**

***Index Terms*—deep learning, autonomous vehicles, adversarial attack, point cloud**

I. Introduction

Transportation is an integral aspect of people's lives; automobiles, more specifically, are indispensable to those residing in the United States. According to the National Safety Council, there were 42,338 automobile related fatalities in 2020 in the United States, which is equivalent to 12.9 deaths per 100,000 people [1]. This number is a significant decrease compared to 1970, in which the United States had 54,633 automobile related deaths, or 26.8 deaths per 100,000 people [1].

This downward trend in fatalities can be attributed to advancements in technology—such as seat belts, cruise control, and, more recently, the introduction of autonomous vehicles. Autonomous vehicles not only contribute to a safer road infrastructure but allow for an increase in mobility and economic stability [2].

While autonomous vehicles offer the prospect of increased safety, they also raise concerns about liability and risk management. Leading 3D object detectors for autonomous vehicles, specifically using the KITTI Dataset have reached, at most, a 96% accuracy on an easy difficulty and a 90% accuracy on a hard difficulty [3]. These detectors have been shown to be vulnerable to adversarial attacks and require further development before deployment.

*Joint first author

We would like to thank the National Science Foundation for funding received under REU Main Site Grant: 1852498- Data Science Research for Healthy Communities in the Digital Age led by PI Prof. Elke Rundensteiner and Co-PI Prof. Chun-Kit Ngan.

Professor Ziming Zhang was supported in part by NSF grant CCF-2006738.

We propose a novel attack methodology which aims to represent naturally occurring noise seen on the road and verify the robustness, or lack thereof, of PIXOR.

A. Related Works

Similar works create novel attack methodology for 3D detectors [5] [6] and implement attack methodology on commonly used datasets [7]. However, there is a absence in validating the robustness of specifically the KITTI dataset using discrete detectors. For instance, the Fast Gradient Sign Method (FGSM) [5] proposes that linearity is the cause of neural networks' vulnerability to adversarial attacks and avoids applying worst-case perturbations to confidently lead to a misclassification by the algorithm. This, however, is not compatible with the PIXOR algorithm, as is used within this paper. PIXOR's input space, while represented by discrete values, uses a container capable of storing real numbers. Attacking PIXOR with FGSM does not simulate natural noise, but instead attempts to continuously change discrete values. Instead of points slightly shifting within the 3D space, the attack shifts a discrete value by a user-defined decimal constant in either the positive or negative direction. This incompatibility between discrete and continuous data produces unreliable metrics and cannot be qualified as an attack which adequately simulates natural noise.

Other attack methods, such as the Joint Gradient Based Attack (JGBA) [6] have demonstrated a similar incompatibility with PIXOR. JGBA aims to break the common defense strategy of removing statistical outliers by optimizing an attack with mathematical consideration to both the original point cloud and its corresponding outlier-removed version. JGBA interacts with the PIXOR detector in a similar mannerism as FGSM; the two attack methods attempt to continuously change a discrete 3D tensor.

There is a lack of robustness verification for KITTI Dataset detectors, as much literature concerning robustness is implemented on datasets unrelated to autonomous vehicles. Datasets such as ModelNet40 and ScanNet40 are often used and have been previously attacked by both FGSM and JGBA [7]. This past work does not extend fully to the realm of autonomous vehicles and must be further explored before full road implementation.

II. Data

A. Dataset

We conducted our experiment with the KITTI Dataset [3], which uses Light Detection and Ranging (LiDAR) to produce a 3D point cloud. The dataset places objects on the road into eight categories: cars, vans, trucks, pedestrians, sitters, cyclists, trams, or miscellaneous. The 7,491 images provided are split into 3,712 and 3,679 points for training and testing, respectively. Each image comprises of at least 10,000 points, Each point is represented by four real number values: the x-y-z coordinates and the reflectance, which is the ratio of light reflected back to the LiDAR sensor. The dataset also provides camera calibration matrices and training labels of the object data set.

B. Data Preparation

Before data can be given to the PIXOR classifier, it must be quantized into a 3D tensor of floating-point values, with dimension sizes 36, 800 and 700 for the z, y, and x dimension respectively. Points are encoded with "1" in the corresponding location in the 3D tensor, and all other tensor elements are set to "0". The increments defined by the tensor indices cannot fully capture the continuous nature of the point cloud, and thus the conversion is inevitably lossy; however, this conversion of loss is consistent among the images.

III. Methodology

Our novel attack incorporates elements of FGSM to create an attack compatible with discrete datasets. We utilize information on the gradient from FGSM to maximize loss to develop a new process that makes slight changes to the positions, not the values, of elements in the 3D tensor.

FGSM's attack adds the signs of the gradient multiplied by a user-defined value to the original data. In contrast, we begin by adding the gradient values themselves to the discrete input data which are notated as tensor values (t). Tensor values are continuous values which range between (-1, 2) (Fig. 1: Gradient Representation (a) Generate t Values). Because of the slight discrepancies and relatively small magnitudes of gradient values in practice, it is sufficient to consider any t value greater than 0.9 to be a point. All other t values contain only the gradient and are not points.

For each point in the tensor, we compare the raw gradient value to the surrounding coordinates' gradients in the 3x3 area. Three of the following scenarios may occur when observing these 26 gradient values:

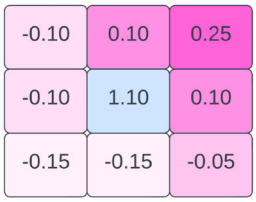
1) All surrounding tensor values are identified to not contain points
2) There is at least one surround tensor value which contains a point
3) The original gradient value is the largest gradient value within the 3x3 area

The initial situation requires we identify the maximum surrounding gradient value and assign "1" said element. The original point is then replace with a "0", finalizing the move (Fig. 1: Gradient Representation (b) Replace Maximum Gradient).

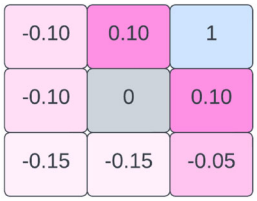
Points which have at least one surrounding tensor value containing a point are not considered. The process then resumes as described in the initial scenario.

If the largest gradient is equivalent to the initial point, the point will maintain its position and not move.

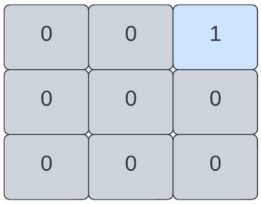
Proceeding all three scenarios, all elements not equal to "1" are set to "0"—which includes the original point (Fig. 1: Gradient Representation (c) Remove Gradients). This ensures that the input is well-formed and will produce reliable metrics.



(a) Generate t Values



(b) Replace Maximum Gradient



(c) Remove Gradients

Fig. 1: Gradient Representation

With this naïve approach, which moves most points, it becomes obvious to the human eye that artificial noise has been introduced when the data is visualized. To better simulate the effects of natural noise, it is necessary to limit the number of points moved and quantify the total change between the original and attacked image.

The measure used to vary the intensity of an attack is identified as the threshold (eq. 1). The threshold measures the percentage change between the maximum gradient value and the center point's gradient value. If the magnitude of this value exceeds that of the specified threshold, the attack will move the point. A smaller threshold allows more points to move and results in a more visually apparent attack while a larger threshold is less visually apparent and moves a fewer number of points.

$$T < \left| \frac{g' - g}{g} \right| \tag{1}$$

T = Threshold
g = Gradient of Original Point
g' = Maximum Gradient (within 3x3 domain)

We ran the experiment with 100 images for 10 different threshold values: specifically, 0.5, 0.75, 1.0, 2.4, 3.0, 3.6, 4.2, 5.0, 10.0, 15.0.

We chose to not implement operations which consider the placement of new points in high-gradient positions or removal of points in negative-gradient positions, as this would create complications when trying to measure and control the severity of an attack.

IV. Results

We measure the robustness of a classifier using metrics and visual similarity between the attacked and original point clouds. Ideal attacks simulate naturally occurring noise and must still be representative of its respective environment.

An effective attack will lead to a misclassification by the algorithm, as represented by a low average precision and high loss while limiting the amount of perturbation to the attacked image. The perturbation can be measured with a qualitative visual analysis of each point cloud and a novel, quantitative metric defined as "points moved per point" (MPP). MPP is a ratio between the number of moves occurred and the total number of points in each image (eq. 2). An implementational nuance makes it possible, while unlikely, for a point to be moved multiple times, so it is important to refer to the ratio as "moves made per point" rather than "points moved per point." In the context of simulating natural noise, we do not find this objectionable.

$$MPP = \frac{m}{p_t} \tag{2}$$

m = Successful Moves
p_t = Total Number of Points

The table below shows the metrics of experiments conducted at each threshold as well as the metrics of the non-attacked 3D detector's testing (Table I: Attack Evaluation Metrics).

Threshold	AP	Precision	Recall	Loss	MPP
Testing	0.6070	0.5860	0.6715	0.2330	0.0000
15.0	0.5463	0.4444	0.6433	0.2079	0.0793
10.0	0.5564	0.4364	0.6520	0.2086	0.1184
5.0	0.5286	0.4096	0.6228	0.2129	0.2180
4.2	0.5096	0.3981	0.6170	0.2144	0.2506
3.6	0.4853	0.3864	0.5965	0.2150	0.2817
3.0	**0.4569**	**0.3660**	**0.5789**	**0.2174**	**0.3211**
2.4	**0.4736**	**0.3734**	**0.5994**	**0.2201**	**0.3708**
1.0	**0.4226**	**0.3327**	**0.5526**	**0.2236**	**0.4943**
0.75	0.2024	0.2259	0.3772	0.2466	0.9089
0.5	0.1783	0.1930	0.3793	0.2334	0.9383

TABLE I: Attack Evaluation Metrics

AP tends to decrease as the threshold decreases, which confirms prior expectations; as more noise is added to a point cloud, its average precision should trend down. There is an inverse relationship between AP and MPP (Fig. 2. Metrics Vs Threshold). MPP relates to the measure of an attack's intensity, where a high MPP will correspond to both a low threshold and a low AP. The testing loss, however, is higher than most of the corresponding loss values of the attacked datasets, which defies expectations. In both these ways, the numeric results may prove inconsistent based on the specific metric used.

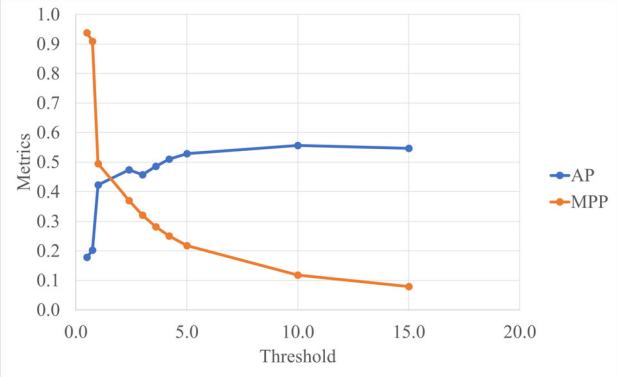


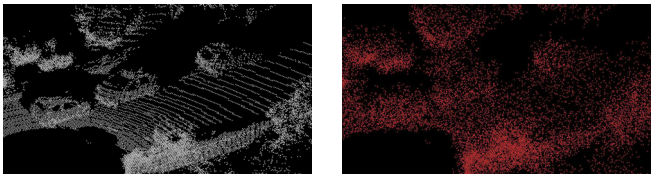
Fig. 2: Metrics VS Threshold

Equally important to the quantitative metrics is the visual feedback of an attack. An ideal attack will replicate noise from the real world, and hence will be unnoticeable to the human-eye. Below are the graphed point clouds of an image attacked at select threshold levels. (Fig. 3: Point Clouds Visualization Based on Threshold). The original point cloud clearly shows 3 cars aligned on the road. These cars are also easily identifiable when the threshold is equal to 10; however this produces an undesirable, higher average precision. Certain thresholds such as 3.0, 2.4, and 1.0 significantly lower the AP without distorting the point cloud beyond human comprehension. Thresholds below 1.0 contain unidentifiable clusters and are generally unsuitable for analysis. Therefore, thresholds between 1.0 - 3.0 are most desirable for attacking a dataset.

V. Limitations

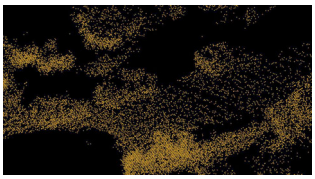
There are limitations and opportunities for further development of our method. In addition to quantized point locations, the 3D input tensors, as provided by the KITTI Dataset, also contain information on the reflectance of points. In each tensor, points are mapped to indices [0-34] along the z dimension, while index 35 contains the average reflectance of all the points within one column. Individual reflectance values for each point are no longer available within the 3D tensor, and we do not fully understand how the reflectance would change when points are shifted. To ensure consistency in the results, we set all the average reflectance values to zero.

The change in points' location is biased. Because our method moves points immediately following detection, moving a point in a certain direction will cause it to be detected and possibly moved again in a future iteration. Movement is

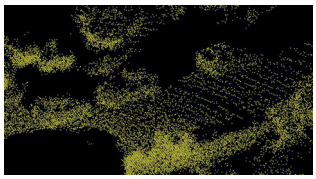


(a) Original, AP = 0.6070

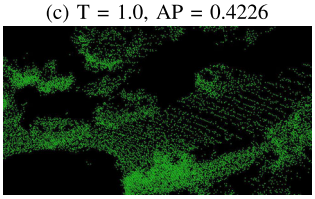
(b) T = 0.5, AP = 0.1783



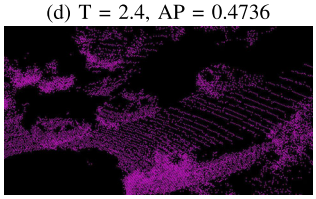
(c) T = 1.0, AP = 0.4226



(d) T = 2.4, AP = 0.4736



(e) T = 3.0, AP = 0.4569



(f) T = 10.0, AP = 0.5564

Fig. 3: Point Cloud Visualization Based on Threshold

therefore biased in the positive x, y, and z direction. Although the bias is not manifest in our visualizations, we are currently unable to quantify how often this occurs and how it affects PIXOR's classification.

Since this attack moves points based on the locally determined percentage change of the gradient (eq. 1), all points are treated as equally important regardless of their magnitude in relation to other points within the global scope. It may be possible to lower PIXOR's AP without increasing the MPP if points are sorted based on their gradients and priority is given to moving the points with the smallest gradient values. This would allow for more precise control of the number of points moved and maximize the loss gained from each iteration.

The above limitations may be contributing to the discrepancies in the results. More work is needed to measure their significance and find solutions.

VI. Conclusion

The robustness of 3D detectors must be verified, especially when concerning autonomous vehicles. PIXOR, a 3D detector used on the KITTI dataset, is not adequately robust and does not successfully defend against adversarial attacks. Other existing attack methods such as FGSM and JGBA do not adequately simulate natural noise on the PIXOR detector and cannot be used to verify robustness. We recommend our novel attack methodology to be integrated on data represented as a 3D tensor array. We anticipate our attack method will aid in making future 3D detectors more robust and fit for real life application. We specifically encourage using a threshold of percentage change in gradients between 1.0 - 3.0, as demonstrated by the 13.34 - 18.44% decrease between training and the respective thresholds.

References

[1] "Historical Fatality Trends.", *National Safety Council*, National Safety Council, https://injuryfacts.nsc.org/motor-vehicle/historical-fatality-trends/deaths-and-rates/ (accessed Jun. 29, 2022).

[2] "Automated Vehicles for Safety." *National Highway Traffic Safety Administration*, United States Department of Transportation, https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety (accessed Jun. 29, 2022).

[3] *The KITTI Vision Benchmark Suite*, Karlsruhe Institute of Technology, 2013. [Online]. Available: http://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=bev

[4] B. Yang, W. Luo, and R. Urtasun, "PIXOR: Real-time 3D Object Detection from Point Clouds," presented at the Conf. on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, Jun. 18-22, 2018.

[5] I. Goodfellow, J. Shlens, C. and C. Szegedy. Explaining and Harnessing Adversarial Examples." presented at the International Conference on Learning Representation, San Diego, CA, USA, 2015.

[6] C. Ma, W. Meng, B. Wu, S. Xu, and X. Zhang. "Efficient Joint Gradient Based Attack Against SOR Defense for 3D Point Cloud Classification." presented at the Association for computing Machinery, New York, NY, USA, 2020.

[7] K. Li, Z. Zhang, C. Zhong, and G. Wang. "Robust Structures Declarative Classifiers for 3D Point Clouds: Defending Adversarial Attacks With Implicit Gradients." presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, June, 2022.